

6060: SISTEMA D'ACCESSIBILITAT PER A ACTES EN DIRECTE

Memòria del Projecte Fi de Carrera
d'Enginyeria en Informàtica
realitzat per
Jordi Paradell Mateo
i dirigit per
Javier Serrano García
Bellaterra, 25 de gener de 2015

El sotasignat, Javier Serrano García
Professor/a de l'Escola Tècnica Superior d'Enginyeria de la UAB,

CERTIFICA:

Que el treball a què correspon aquesta memòria ha estat realitzat
sota la seva direcció per en Jordi Paradell Mateo

I per tal que consti firma la present.

Signat: Javier Serrano García
Bellaterra, 25 de gener de 2015

Índex

1 Introducció.....	9
El problema.....	11
Objectiu.....	12
Viabilitat.....	13
Planificació.....	14
Organització del document.....	15
Context sociopolític.....	15
Àmbit legislatiu.....	17
2 Plataforma de desenvolupament.....	19
Qt.....	20
Android.....	21
Reconeixement automàtic de la parla.....	23
Control de versions.....	24
3 Anàlisi de requeriments i funcionalitats.....	25
Servidor.....	26
Client mòbil.....	26
Programa de projecció.....	27
4 Disseny i implementació del sistema.....	29
Comunicacions.....	29
Servidor Broadcaster.....	31
Client Display.....	34
Client d'Android.....	35
Protocol.....	37
5 Conclusions i ampliacions.....	39
Bibliografia.....	41

Índex de figures

Figura 1: Sala d'actes del Rectorat de la UAB.....	9
Figura 2: Gran Teatre del Liceu.....	11
Figura 3: Planificació temporal del projecte, Diagrama de Gantt.....	14
Figura 4: Qt Creator 2.8.1.....	20
Figura 5: Eclipse SDK 4.2.2 Juno amb diversos plugins.....	21
Figura 6: Eclipse en la perspectiva d'Android/Java.....	22
Figura 7: Esquema de funcionament d'un reconeixedor automàtic de la parla.....	23
Figura 8: Sala de control de la sala d'actes del rectorat.....	29
Figura 9: Diagrama físic del sistema.....	30
Figura 10: Servidor Broadcaster, programa principal del sistema.....	31
Figura 11: Wireshark, eina bàsica en la comprovació del funcionament en xarxa.....	33
Figura 12: Display, el programa per a projectar els subtítols a una sala o capturar-los.....	34
Figura 13: Funcionament del programa dins de la sala.....	35
Figura 14: Programa d'Android Infoaccessibilitat UAB.....	36
Figura 15: Protocol de reconeixement d'idiomes oferts.....	37
Figura 16: Demostració de funcionament al gerent de la UAB al rectorat.....	39

1 Introducció

Aquesta memòria del projecte final de la carrera d'Enginyeria Informàtica anomenat «Sistema d'accessibilitat per a actes en directe», és fruit d'un treball continuat de pràctiques en el servei d'infoaccessibilitat de la UAB (InfoAcc-UAB)¹, concretament dins de la part corresponent amb el Departament de Telecomunicació i d'Enginyeria de Sistemes.



El servei d'infoaccessibilitat s'emmarca dins del pla estratègic de la UAB que vol garantir la igualtat d'oportunitats de les persones amb diversitat funcional i té com a objectiu fer que un alumne amb dificultats visuals, auditives o lingüístiques no deixi de poder rebre la informació que reben els seus companys en qualsevol aula o sala d'actes de la UAB.

Actualment la UAB disposa de cabines aïllades a sala d'actes del rectorat i a l'aula magna de la facultat de traducció i interpretació per a poder desenvolupar-hi tasques de traducció i interpretació.



Figura 1: Sala d'actes del Rectorat de la UAB

¹ Servei d'infoaccessibilitat. InfoAcc-UAB, <http://serveis.uab.cat/infoacc/>

L'objectiu final del servei és crear una unitat mòbil d'accessibilitat per a donar servei a tots els campus de la universitat (Sabadell, Bellaterra i Barcelona) i als demés que ho puguin sol·licitar. Per a aconseguir-ho, l'equip de la unitat mòbil s'haurà d'adaptar a les circumstàncies que es presentin en cada sala tot mantenint uns elements comuns.

La part variable ve causada per la diversitat d'instal·lacions i situacions, tant físiques com lògiques que podem trobar-nos-hi dins, com ara la presència o no d'un projector, d'una cabina aïllada, de bucles magnètics (tecnologia per a audiòfons), càmeres, emissió a Internet en directe, intèrpret de llengua de signes, etcètera. És per això que el desenvolupament del sistema ha estat força relacionat amb els responsables d'audiovisuals de les sales.

Aquest projecte té com a objectiu el disseny i implementació de la part que podem considerar comuna a tots ells, un sistema que anomenarem UAS (Universal Accessibility System) i que proporciona un sistema de baix cost, fàcil transport i desplegament per a difondre els continguts accessibles als usuaris d'aquest servei.

A la hora de parlar de continguts accessibles, els elements més comuns i funcionals són el subtítols i les audiodescripcions. Tanmateix, a la hora de generar els subtítols per a un mitjà tant habitual com la televisió, podem fer servir la següent taxonomia de la Corporació Catalana de Mitjans Audiovisuals (CCMA)², on es defineixen quatre categories en funció dels recursos dels que disposen a la hora de generar-los:

- Subtítols diferits
- Subtítols en directe amb preparació prèvia
- Subtítols sense preparació prèvia fent-ne un resum
- Directes amb transcripció literal

La primera categoria no entra dins dels objectius del servei ja que les ponències i classes magistrals que es puguin donar sempre seran en directe, per tant, ens enfocarem en les tres altres categories.

2 Servei de subtitulació, CCMA, <http://www.ccma.cat/atencio/subtitulacio/>

El problema

El problema que intenta solucionar aquest projecte pot passar desapercebut per a la majoria de persones, però un cop ens hi fixem, trobarem que existeixen inconvenients reals per a que les persones amb diversitat funcional puguin accedir a la informació produïda en el mateix acte.

Els darrers canvis legislatius obliguen a moltes sales de teatre, d'actes i de cultura en general a disposar de servei d'infoaccessibilitat, i aquest problema fins ara era mitigat amb solucions poc econòmiques i versàtils amb empreses que muntaven un sistema propietari i tancat.

Un cas evident és el Gran Teatre del Liceu de Barcelona, on es disposen pantalles al darrera de les butaques, o altres sales on es projecten super-títols sobre l'òpera o sistemes més avançats on es projecten els continguts accessibles al darrera dels espectadors (de forma invertida) i aquell que ho desitgi pot girar un mirall per a llegir-los, permetent als demés no haver-los de veure.



Figura 2: Gran Teatre del Liceu

En el document referenciat d'Estela Oncins es fa tot un estudi dels diversos sistemes existents en el mercat i es constaten forces mancances en tots ells, com ara els costos econòmics, la incapacitat d'emetre audiodescripcions simultàniament amb els subtítols, múltiples idiomes, etc.

Avui dia podem dir que els dispositius *smartphone* i tauletes, amb més d'un 80% d'índex de penetració a Espanya en les persones menors de 44 anys³ són omnipresents i estan aquí per a quedar-se, per tant podem assumir que una gran part del públic disposa d'un dispositiu portàtil capaç de rebre dades i mostrar-les per una pantalla, just el que necessitem per ensenyar subtítols o idiomes a les persones amb dificultats auditives o lingüístiques.

Per a les persones amb dificultats visuals, actualment els seus dispositius tenen eines per a l'accessibilitat i la més fonamental són les capacitats de TTS (*text-to-speech*) en múltiples idiomes, que els guien pels menús del sistema operatiu i de les aplicacions i ens permetrà generar audiodescripcions al vol.

És per això que considero a les tauletes i els smartphones com a dispositius ideals per a aquestes situacions.

Per a generar els subtítols necessitarem un programa amb una interfície gràfica senzilla, robust i que pugui mostrar amb facilitat el concepte que el servei proveir.

Si l'operari de l'aplicació, generalment un professional de la interpretació, manca de velocitat d'escriptura amb el teclat, el servei l'oferirà les capacitats de «reparlat» per a que pugui fer servir la veu per a introduir la informació.

Objectiu

Per tant, els objectius fonamentals que aquest projecte vol assolir són:

- Dissenyar i implementar un sistema que permeti distribuir subtítols i audiodescripcions a *smartphones* i tauletes dels usuaris així com a les pantalles de les sales.
- Poder emetre continguts prèviament preparats i subtítols generats en directe en un mateix programa.
- Facilitar el «reparlat», o sigui, dictar oralment de forma resumida els subtítols a l'ordinador per a extreure'n el text, poder-lo editar i enviar per tal d'accelerar el procés.
- Permetre la càrrega de fitxers en format estàndard (SRT) en els casos en que es disposa de discursos preparats.

3 Our Mobile Planet, http://think.withgoogle.com/mobileplanet/en/graph/?country=es&category=DETAILS&topic=Q00&stat=Q00_1&wave=2013&age=a1&age=a2&age=a3&gender=all&chart_type=&active=age

- Ha de ser compatible amb l'ordinador de la sala de control del rectorat, o sigui, funcional en Windows XP i amb el portàtil del servei que funciona també amb Ubuntu.

Viabilitat

El concepte de retransmetre subtítols des d'una estació base a un conjunt d'smartphones a través de Wi-Fi no suposa cap dificultat tècnica, es té planejat utilitzar les capacitats de Broadcast de datagrames UDP del protocol IPv4, i els problemes més evidents són la pèrdua de paquets en un entorn tant hostil com és l'espectre lliure de 2.4Ghz. L'estàndard 802.11 estableix sistemes de Multicast/Broadcast en capa física i per tant no s'espera cap tipus de congestió per l'ús dels programes en la xarxa local.

Moltes sales més grans ja disposen actualment de connexió sense fils a la xarxa i per tant no considerem els problemes de cobertura de les mateixes. En aquests casos s'haurà de modificar les adreces IP del servidor i del programa per tal de compatibilitzar-les amb la nova xarxa ja que el projecte ha estat pensat per a funcionar només en la nostra pròpia xarxa portàtil on controlem les direccions nosaltres mateixos.

Econòmicament, només requerim d'un portàtil amb els programes adients per a generar i transmetre els subtítols, mitjançant Ethernet o sense fils, un encaminador/punt d'accés, que en el nostre cas serà un Linksys WRT54GL. Els usuaris podran descarregar el programa mòbil des de la seva tenda d'aplicacions i les sales podran facilitar una petita quantitat de tauletes de molt baix cost als usuaris que així ho requereixin, fent d'aquest un sistema amb un cost de maquinari inferior als 1000€.

Com és de suposar, els requeriments de hardware de l'ordinador són mínims i s'ha pensat en tot moment en que el software funcioni en entorns lliures i amb pocs recursos. Tot i fer servir una xarxa 802.11g, els 54Mb/s són exageradament ràpids per a l'ús que l'hi volem donar i els dispositius mòbils tampoc necessiten grans especificacions tècniques.

Planificació

La planificació temporal del projecte es detalla en el següent diagrama de Gantt, és important remarcar que la dedicació estimada és d'unes quatre hores diàries.

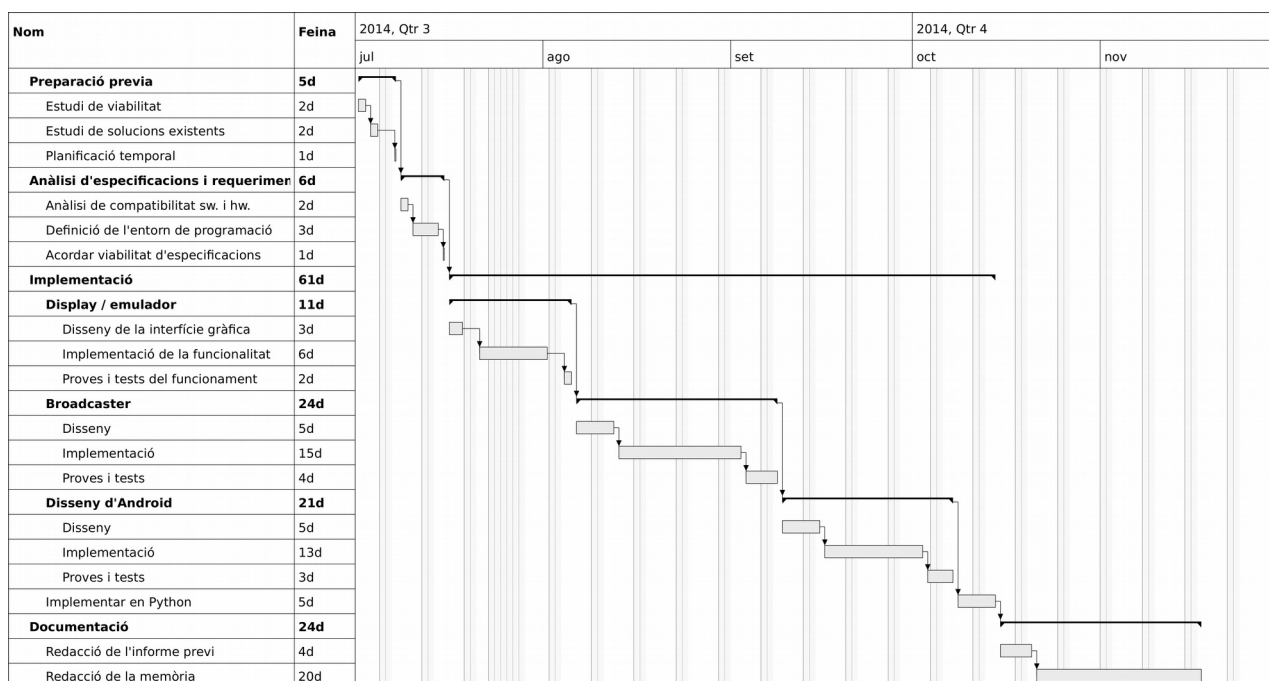


Figura 3: Planificació temporal del projecte, Diagrama de Gantt

En una primera fase s'ha estudiat la viabilitat d'aquest projecte, analitzant els requeriments, solucions preexistents, material del que es disposa i les diverses formes de portar a terme una solució al problema inicial.

En una segona fase s'han anat realitzant cada una de les aplicacions necessàries per al funcionament del sistema, deixant l'apartat de la documentació per a una tercera fase.

Finalment hem deixat pel final la conversió del codi a Python ja que s'ha programat en C++ al ser el llenguatge original de Qt, és molt més popular i es disposa de molta més informació. Malgrat això, el manteniment del programari, una fase molt important del cicle de vida de les aplicacions, es vol realitzar en Python i és per això que ho traduirem i entregarem en PyQt.

Tot el desenvolupament s'ha anat seguint setmana rere setmana per part del director del mateix projecte i s'ha intentat en tot moment tenir versions parcials funcionals per tal d'afegir o treure característiques a les aplicacions, sobretot per a les dates dels assajos amb personal extern a l'escola.

Organització del document

- En el següent capítol d'aquest document es mostra l'entorn on s'ha desenvolupat tot el projecte per tal de facilitar la reproductibilitat i qualsevol posterior millora, modificació o mer manteniment.
- Al tercer capítol es llisten els requeriments dels sistema els quals definiran la forma i funció de l'aplicació i de les comunicacions, tot entenent que s'ha optat per un disseny iteratiu sobre una base mínimament funcional i no d'un disseny inicial complet i definitiu, fet que explica diverses ineficiències en el disseny de l'arquitectura del sistema.
- A continuació es detalla el funcionament de cada una de aplicacions per tal d'explicar com s'ha dissenyat, creat i com l'usuari hi pot interactuar, servint de petit manual. S'exposen els requeriments a mesura que s'han anat completant i es mostren les captures de pantalla de la interfície d'usuari i diagrames del protocol de comunicació.
- Acabem amb les conclusions, analitzant el compliment dels objectius i s'ofereixen diverses idees per a l'ampliació del projecte
- Finalment, llistem les referències bibliogràfiques mentre que les referències de caire més puntual es presenten com anotacions al peu de pàgina. El resum del projecte s'ofereix a la contraportada en tres idiomes.

Context sociopolític

Després de dues grans guerres mundials i diverses revolucions tecnològiques, el món s'ha anat transformant ràpidament i acords plurinacionals com la declaració universal dels drets humans es preocupen per a corregir errors del passat, declarant principis fonamentals com ara la dignitat, la llibertat, la igualtat, la fraternitat i la no discriminació.

Des de l'any 1991, s'estan fent lleis per a obligar als diferents òrgans, tant públics com privats, a eliminar múltiples barreres arquitectòniques que impedièren que les persones amb un o d'altre tipus de discapacitat o minusvalidesa (diversitat funcional) poguessin ésser tractats amb igualtat de condicions amb aquelles que no les patien. Aquestes lleis es concentraven en les barreres físiques i deixaven de banda l'accés l'accés universal a la informació i la cultura, que defineixen quasi bé tots els aspectes socials d'avui dia.

Aquest projecte representa la voluntat de la societat actual per a no discriminar i fer accessible una part de la informació que fins ara no ho era. Ara estem a l'era de l'espai i la informació, i la carrera per la qual presento aquest projecte és un fidel reflex de l'època en que vivim.

L'accessibilitat per a actes en directe porta anys sent un objectiu per la nostra societat ja que permet la igualtat de tothom en diferents tipus d'espectacle que fins fa relativament poc temps semblava impossible pensar que es pogués arribar a garantir.

Tot i que s'han fet estudis i alguns avenços en aquesta matèria, no s'arribava a concretar amb dispositius que funcionessin correctament. D'aquí parteix la meva motivació per fer aquest projecte d'utilitzar smartphones per permetre aquesta accessibilitat amb unes garanties mínimes com que és de baix cost, fàcil d'utilitzar i d'implementar i per tant, assegura l'accés d'aquesta tecnologia a tothom.

Un altre aspecte motivador important és que aquest multidisciplinari que m'ha permès col·laborar amb altres professionals de diferents àmbits com la traducció i interpretació, telecomunicacions o multimèdia, fent així possible que pogués treballar en un grup que contempla altres àmbits que no són la informàtica i aprenguéssim com està l'accessibilitat cultural realment, volent aportar un treball que pugui ser útil.

També, m'ha permès adquirir nous coneixements en el meu propi marc disciplinar com dissenyar i crear interfícies multi plataforma i fer aplicacions per dispositius mòbils que avui en dia amb l'increment de la quota de mercat de la tecnologia Android són fonamentals pel nostre entorn professional i per l'avenç de la implantació de la tecnologia en nous sectors. He pogut aprofitar per aprendre nous *frameworks* i tècniques de programació que de ben segur em seran útils en un futur.

Àmbit legislatiu

La llei d'accessibilitat de 1995, pionera llavors a Catalunya, permetia la supressió de barreres arquitectòniques i garantia l'accés a tothom en els edificis públics i privats de nova construcció; però, tot i suprimir les barreres físiques, no es va anar més enllà per incloure també l'àmbit cultural, ja que llavors amb les tecnologies existents semblava una fita impossible, per fi, quasi vint anys més tard, clarament aquesta llei quedava obsoleta i es va preparar una nova llei que va ésser publicada el dia 4 de novembre d'aquest any al DOGC.

La llei d'accessibilitat 13/2014⁴ deroga l'anterior llei de 1995, manté la supressió de barreres arquitectòniques, però aquest cop, inclou el deure de qualsevol organització a garantir la igualtat en els diferents àmbits culturals entre ells, els espectacles en directe, és per això, que aquest projecte vol permetre complir la llei i oferir una solució dins de la situació econòmica actual que tingui un baix cost per aquestes organitzacions així com pels usuaris que ho facin servir ja que és una tecnologia mòbil i permeten que la puguin utilitzar arreu on vulguin anar, fent també que sigui possible la pròpia autonomia i llibertat personal esmentada a la declaració universal dels drets humans.

4 Llei 13/2014, del 30 d'octubre, d'accessibilitat. <http://portaldogc.gencat.cat/utillsEADOP/PDF/6742/1379017.pdf>

2 Plataforma de desenvolupament

Durant tot el desenvolupament del programari s'ha fet servir software completament lliure⁵, des del sistema operatiu, els entorns de programació, les llibreries, les eines i utilitats, fins al software per a l'edició d'aquest mateix document.

L'UAS està pensat per a ser una plataforma mòbil amb un punt d'accés Wi-Fi (WRT54GL) i un portàtil amb arranc dual (Windows 8 i Ubuntu), però el rectorat de l'escola només disposa d'ordinadors amb Windows XP, fet que implica que la compatibilitat entre diverses versions de Microsoft Windows i diversos sistemes operatius sigui d'extrema importància.

Per a la generació i transmissió dels continguts farem servir un ordinador personal genèric mentre que s'ha decidit per a la plataforma Android d'*smartphones* i tabletas com a mitjà mòbil, tant per la seva popularitat com pel seu baix cost de desenvolupament.

Per a programar el software i crear una interfície gràfica intuïtiva, s'ha decidit fer servir Qt, basat en C++. I pels dispositius mòbils Android es fa servir el framework original d'Android, basat en Java.

L'entorn que s'ha fet servir per a desenvolupar tot el projecte consisteix en una distribució Gentoo Linux de 64 bits amb KDE 4.14.3 com a gestor de finestres sobre l'entorn gràfic X.Org X Server 1.15.2 i Linux 3.16.5 com a kernel, aquest document ha estat redactat en LibreOffice Writer 4.2.8.2.

Per a poder fer funcionar el software de forma portable als entorns Microsoft, la compilació del codi ha hagut d'incloure les llibreries Qt de forma estàtica, generant un executable de més d'onze MiB, però la versió final que es publica amb aquesta memòria funcionarà amb Python i PyQt i per tant, aquest punt deixa de tenir importància.

Les proves del software s'han realitzat en Microsoft Windows[®] 7, Windows[®] 8, Mac OS[®], Ubuntu i Debian.

5 La definició de Programari Lliure – Projecte GNU – Free Software Foundation (FSF) | <https://www.gnu.org/philosophy/free-sw.html>

Qt

El *Qt framework*, pronunciat «cute» en anglès, és una biblioteca multiplataforma per a desenvolupar aplicacions amb interfície gràfica d'usuari (GUI) creada a l'any 1992 i alliberada a l'any 2000 per Trolltech. Basat en l'estàndard C++, és una eina molt provada i amb gran quantitat de recursos online per a ajudar a programadors de qualsevol nivell.

S'ha escollit aquest software pels següents motius:

- Qt Permet aprofitar el mateix codi per a funcionar en més de 15 plataformes diferents⁶
- Integració nativa amb l'entorn d'escriptori (KDE)
- Llicència de codi lliure GPL 3.0
- Experiència prèvia realitzant aplicacions gràfiques en *OpenGL* i *Shaders* a Gràfics per Computador II
- Gran quantitat d'informació a la xarxa al ser tant popular.
- L'IDE té la documentació i referència integrades.
- *Bindings* a Python amb PyQT o PySide

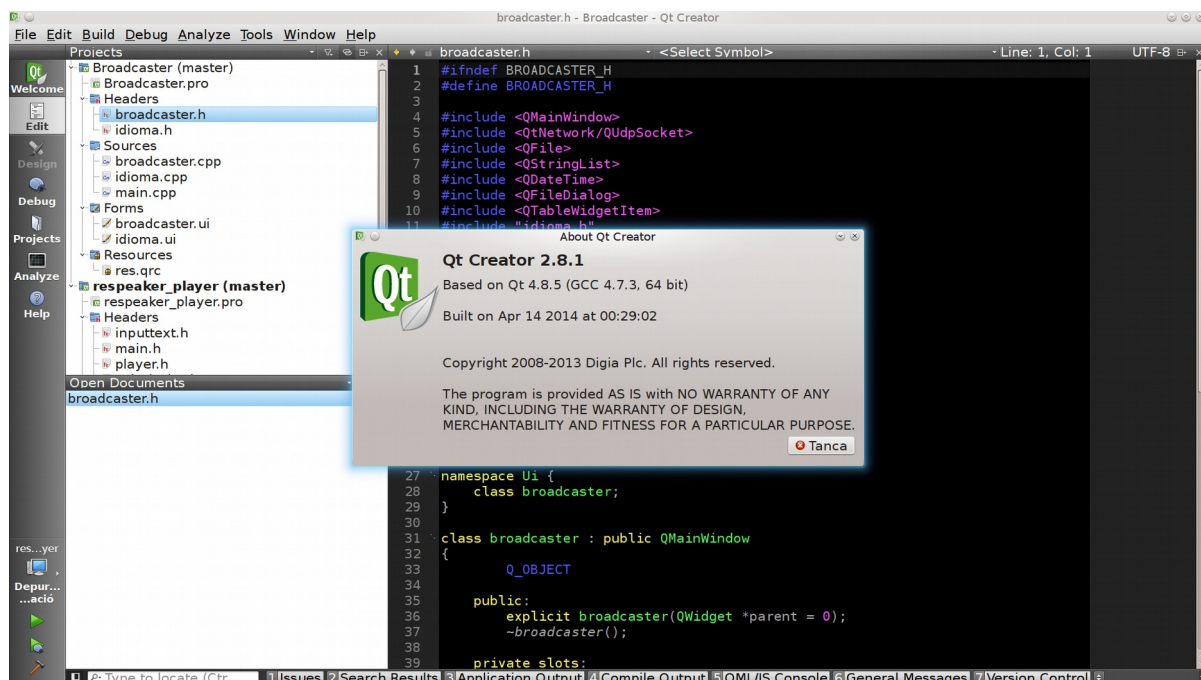


Figura 4: Qt Creator 2.8.1

⁶ Supported Platforms | Qt 5.4, <http://doc.qt.io/qt-5/supported-platforms.html>

L'IDE que s'ha escollit per a desenvolupar el programari és el Qt Creator 2.8.1 compilat amb el GCC 4.7.3 que proporciona totes les eines necessàries per a un desenvolupament molt còmode. La versió de Qt tant per a compilar l'entorn IDE com per a compilar el nostre software és la Qt 4.8.5 i el nostre programari el compilem amb una versió més recent, el GCC 4.8.3. En tots els casos parlem d'arquitectura x86_64 (64 bits).

Una vegada comprovat el funcionament de l'aplicació en C++, s'ha passat a programar la seva traducció en Python, fent servir l'Eclipse amb el plugin PyQt com a IDE.

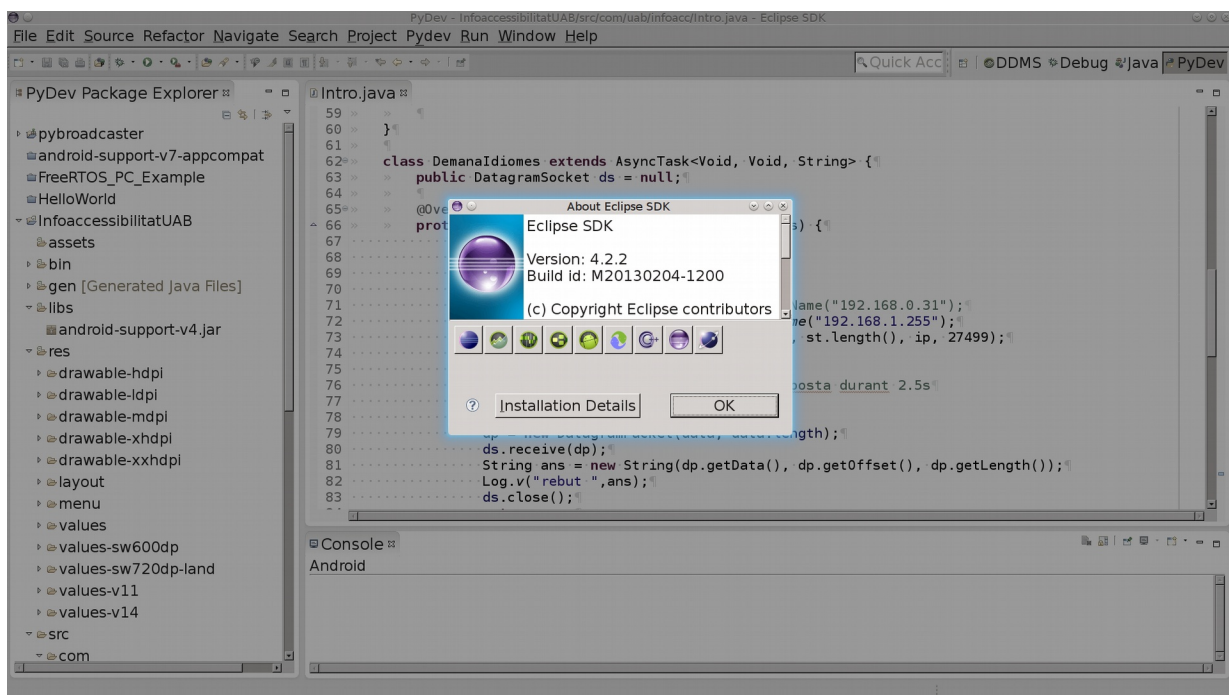


Figura 5: Eclipse SDK 4.2.2 Juno amb diversos plugins

Fulla d'estil: Tot el codi font està sagnat amb tabuladors, a raó de 4 espais per tabulador i amb un límit horitzontal de 100 caràcters.

Android

Per al dispositiu mòbil també podíem haver fet servir Qt, però havent pres part dels cursos de programació en Android oferts per la UAB, resulta més avantatjós posar en pràctica els coneixements adquirits per a programar amb l'entorn natiu.

Motius:

- Experiència prèvia al realitzar els cursos de lliure elecció del dEIC per a programar en Android
- Programació en el llenguatge oficial d'aplicacions d'Android
- Llicència de codi lliure
- Gran quantitat d'informació a la xarxa al ser l'estàndard.
- Assistents de publicació a Google Play

Atès que el client és molt diferent del servidor, no es reutilitza gaire codi i per tant s'ha optat per a fer servir l'Android Software development kit (Android SDK o ADK).

Aquest SDK es pot descarregar de la pàgina d'Android com un IDE complet que no cal instal·lar, només executar, l'entorn de desenvolupament és el ben conegut Eclipse que s'ha anat fent servir durant tota la carrera i accepta multitud de llenguatges de programació diferents com a *plugins* del mateix.

Una alternativa a descarregar-se tot l'IDE consisteix en tenir una instal·lació d'Eclipse en el nostre ordinador i baixar-se només el *plugin* ADT (*Android Development Tools*) per tal de no tenir múltiples instal·lacions d'Eclipse redundants al sistema.

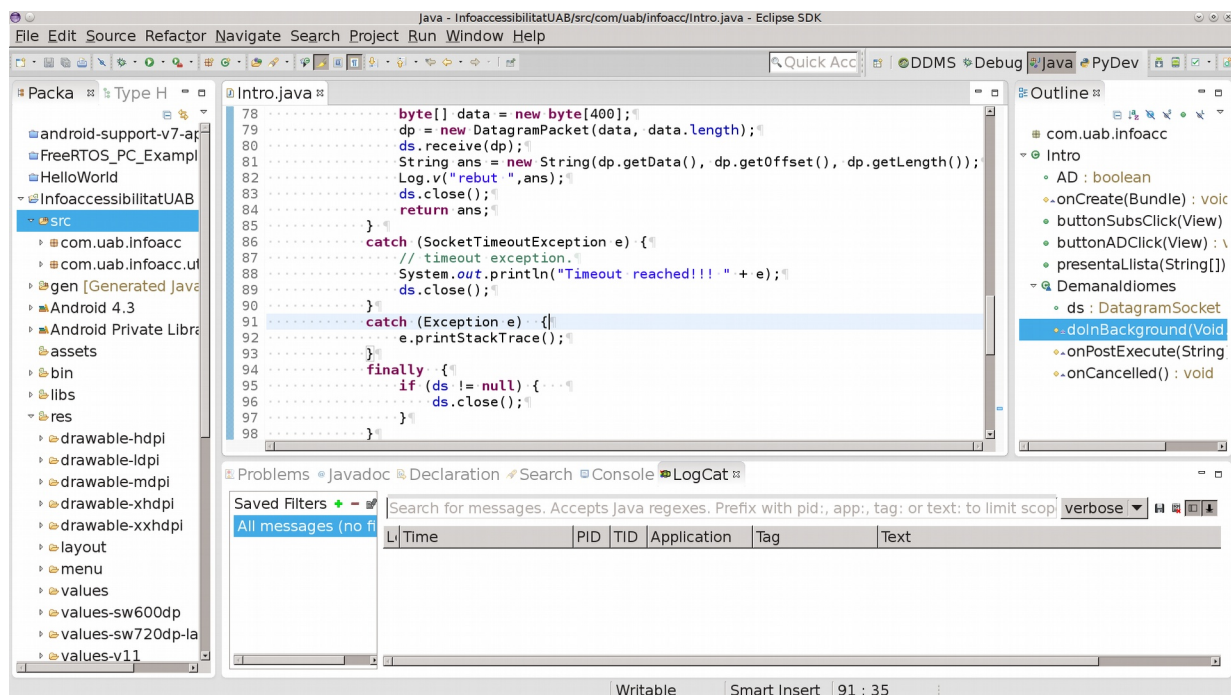


Figura 6: Eclipse en la perspectiva d'Android/Java

A la captura de pantalla es pot observar com l'Eclipse té multitud de plugins instal·lats, tant per a programar en C, C++, Java, Android, Python, PHP, etc. En el nostre cas, fem servir l'Eclipse Juno 4.2.2 i per a executar el codi disposem d'entorns virtuals que emulen a un smartphone o tauleta i també de connexió de «*Dalvik Debug Monitor Server (DDMS)*» a una tauleta Google Nexus 7 i a un mòbil Samsung Galaxy Note 3.

Reconeixement automàtic de la parla

Finalment, per a facilitar i accelerar la introducció del text a l'ordinador proporcionem dues solucions, dins l'entorn de Microsoft Windows disposem d'un software de reconeixement de la parla (Automatic Speech Recognition) anomenat Dragon NaturallySpeaking⁷, que en la seva versió 12.5 que ens permet reconèixer anglès, francès, alemany, italià, castellà, holandès i japonès. El text reconegut s'introdueix automàticament on tinguem el cursor.

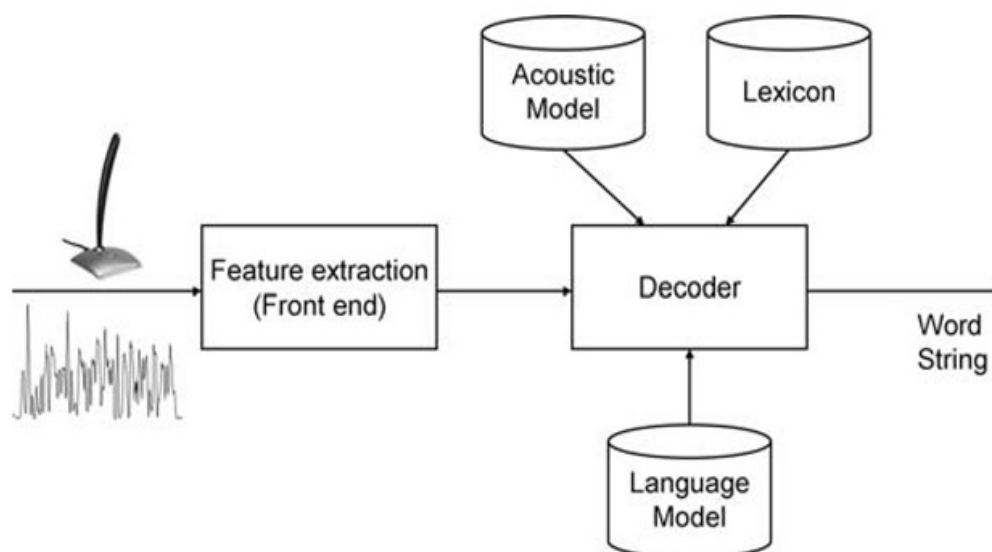


Figura 7: Esquema de funcionament d'un reconeixedor automàtic de la parla

Per a poder processar el català, farem servir una alternativa més complexa, es tracta d'un programa basat en un software de construcció i manipulació de cadenes ocultes de Markov anomenat *Hidden Markov Model Toolkit (HTK)* i la seva API (*The ATK Real-Time API for HTK*) ajustat al català ara farà uns anys al mateix departament de la UAB.

⁷ Dragon – Dragon NaturallySpeaking – Nuance, <http://www.nuance.com/dragon/index.htm>

Aquest software, desenvolupat al Laboratori d'intel·ligència computacional del departament d'enginyeria de la universitat de Cambridge està disponible sota una llicència especial i malgrat ser multi plataforma, la única versió que disposem és una modificació de l'original que només funciona entorn GNU/Linux, és per això que quan executem el nostre software en un entorn Microsoft Windows fem servir una màquina virtual amb un entorn de GNU/Linux (Debian) per a fer funcionar l'ASR en català.

Per a introduir el so fem servir amb uns auriculars amb micròfon USB els quals fem un *pass-through* del host al client. Una vegada reconegut el missatge, aquest retorna la informació a través d'un datagrama UDP en format XML al nostre servidor.

Control de versions

Per tal de fer un seguiment de la feina realitzada, disposar de còpies de seguretat, recuperar codi de versions anteriors i permetre sincronitzar múltiples estacions de treball, es fa servir un servei de control de versions. En aquest cas es fa servir Git i com a servidor remot es fa servir Bitbucket⁸.

Bitbucket ofereix repositoris privats il·limitats per a grups de treball de menys de 5 persones i, després de fer-lo servir també a Gràfics per a computador II, considero que és la solució gratuïta de còpies de seguretat i control de versions més adient per al desenvolupament.

Per a autenticar-se al repositori i pujar i baixar el codi d'una forma molt més àgil i segura, s'ha optat per a fer servir criptografia de clau pública amb claus RSA-2048.

8 Free source code hosting for Git and Mercurial by Bitbucket, <https://www.bitbucket.org/>

3 Anàlisi de requeriments i funcionalitats

El sistema disposarà de dos actors implicats:

- Traductor i/o intèrpret (professional)
- Usuari del servei

Els casos d'ús que ens podem trobar es separen en funció de si tenim el discurs, text o ponència amb antelació a l'esdeveniment o no. En el cas afirmatiu, el professional haurà de crear un fitxer de subtítols i audiodescripcions per a cada idioma que desitgi mostrar.

Tant si el professional disposa dels continguts preparats, haurà de ser present en una cabina de la sala per tal de gestionar quan s'emeten els continguts infoaccessibles i estar preparat per a qualsevol improvisació que es produeixi.

Els usuaris del servei seran els espectadors de l'esdeveniment que mitjançant el gestor de programari dels seus dispositius mòbils podran descarregar-se l'aplicació del servei i una vegada connectats a la xarxa sense fils local i seleccionat l'idioma desitjat, puguin rebre els continguts emesos.

El sistema té tres requeriments funcionals principals:

- S'ha de poder transmetre subtítols i audiodescripcions a través d'una xarxa sense fils informàtica a dispositius mòbils.
- Els continguts infoaccessibles s'han de rebre en dispositius Android i mostrar als usuaris, tant de forma escrita com parlada.
- El format de fitxer ha de ser SubRip Text⁹ (*.srt) amb lleus modificacions per a acceptar les audiodescripcions.

A continuació es llisten els requeriments tant funcionals com no funcionals de cada una de les tres programes en que s'ha dividit el sistema.

9 SRT Subtitles | Matroska, <http://www.matroska.org/technical/specs/subtitles/srt.html>

Servidor

- El programari d'emissió de subtítols ha de ser multi-plataforma, és a dir, funcionar com a mínim en Microsoft Windows, en GNU/Linux i MacOS.
- S'ha de poder carregar fitxers de subtítols i audiodescripcions en múltiples idiomes i poder eliminar-los quan es necessiti.
- En el cas de tenir els continguts prèviament preparats i traduïts, s'han de poder emetre simultàniament a tots els espectadors.
- S'ha de disposar un registre dels subtítols emesos en un idioma per a una possible postproducció.
- La interfície d'usuari del servidor ha de ser fàcil d'utilitzar per a qualsevol persona i disposar d'accessos directes de teclat per tal d'agilitzar les operacions.
- Qualsevol interacció amb el programa ha d'impedir que es perdi el focus del camp d'introducció de text.
- El programa ha de permetre ajustar la longitud de les línies de subtítols d'una forma fàcil.
- Ha de ser compatible amb reconeixadors automàtics de la parla per a agilitzar la introducció del text
- Ha de funcionar amb el reconeixedor de parla en català que transmet els resultats a través de la xarxa en UDP.

Client mòbil

- L'usuari ha de disposar de l'aplicació en l'idioma del seu *smartphone*.
- S'ha de poder escollir l'idioma en que es vol rebre els continguts retransmesos entre l'oferta d'idiomes proporcionada per l'operari.
- Triar entre subtítols, audiodescripcions, i subtítols llegits (funcionalitat de traducció simultània)
- L'aplicació ha d'estar ben integrada amb l'ecosistema d'aplicacions d'Android-

- La pantalla on es mostrin els subtítols ha de ser fosca amb la lletra groga, impedit la personalització per part de l'usuari.
- L'aplicació s'ha de poder descarregar a través del sistema de Google Play
- L'aplicació només ha de mostrar els idiomes que estiguin disponibles a l'esdeveniment.
- S'ha de fer servir un port de xarxa diferent per idioma per tal d'intentar conservar la compatibilitat amb programes anteriors del servei.

Programa de projecció

- Haurà de ser multiplataforma per tal d'adaptar-se a cada client diferent.
- La finestra de projecció s'haurà de poder disposar en pantalla completa sense marges fàcilment.
- La finestra on es mostren els continguts haurà d'estar per sobre de qualsevol altre per a minimitzar incidències.
- Els colors i tipografia podran ser ajustats per l'operador.
- En cas d'haver de mostrar un text diferent del de l'esdeveniment, s'han de poder escriure missatges personalitzats en pantalla per a la sala.
- L'operari podrà seleccionar l'idioma que vol que es mostri a la finestra

4 Disseny i implementació del sistema

Comunicacions

El departament disposa d'un encaminador i punt d'accés 802.11g Linksys WRT54GL que serà la capa física on les comunicacions sense fils es duran a terme. Aquest encaminador *ofereix una xarxa Wi-Fi sense autenticació ni xifrat* per a que qualsevol usuari s'hi pugui connectar.

El dispositiu es pot connectar a la xarxa de la UAB a través del seu port WAN i mitjançant una adreça MAC modificada. D'aquesta forma podem col·locar l'encaminador entre l'ordinador de la sala de projecció i la seva connexió de xarxa, de tal forma l'ordinador pertanyi a la mateixa subxarxa i pugui rebre els continguts per a poder-los projectar o retransmetre a Internet.

A la següent imatge es pot observar tot l'equipament que es fa servir al rectorat, a l'esquerra hi ha un portàtil que mostra l'*streaming* infoaccessible de la pàgina web de la UAB, l'ordinador de sobretaula amb dues pantalles composa la senyal de vídeo amb dos fonts simultànies de subtítols que rep del servidor (el portàtil amb pantalla blanca de la dreta) i amb un servei extern d'estenotípia rebut a través d'internet.



Figura 8: Sala de control de la sala d'actes del rectorat

El servidor s'executa en el portàtil de davant de la cadira i envia els continguts a l'encaminador situat a sobre la torre de l'ordinador, i aquest al mateix temps, fa el Broadcast de la senyal fins a la tauleta que es troba a la cantonada de la taula, representant a l'espectador. El portàtil situat a la dreta del tot és el que s'ha fet servir per a desenvolupar el projecte i es pot observar amb el Qt Creator obert.

El servidor emetrà missatges UDP a l'adreça de *Broadcast* d'una xarxa WLAN des d'un ordinador, generalment portàtil, per ser rebuda per a dispositius Android amb el software descarregat de la Play Store de Google.

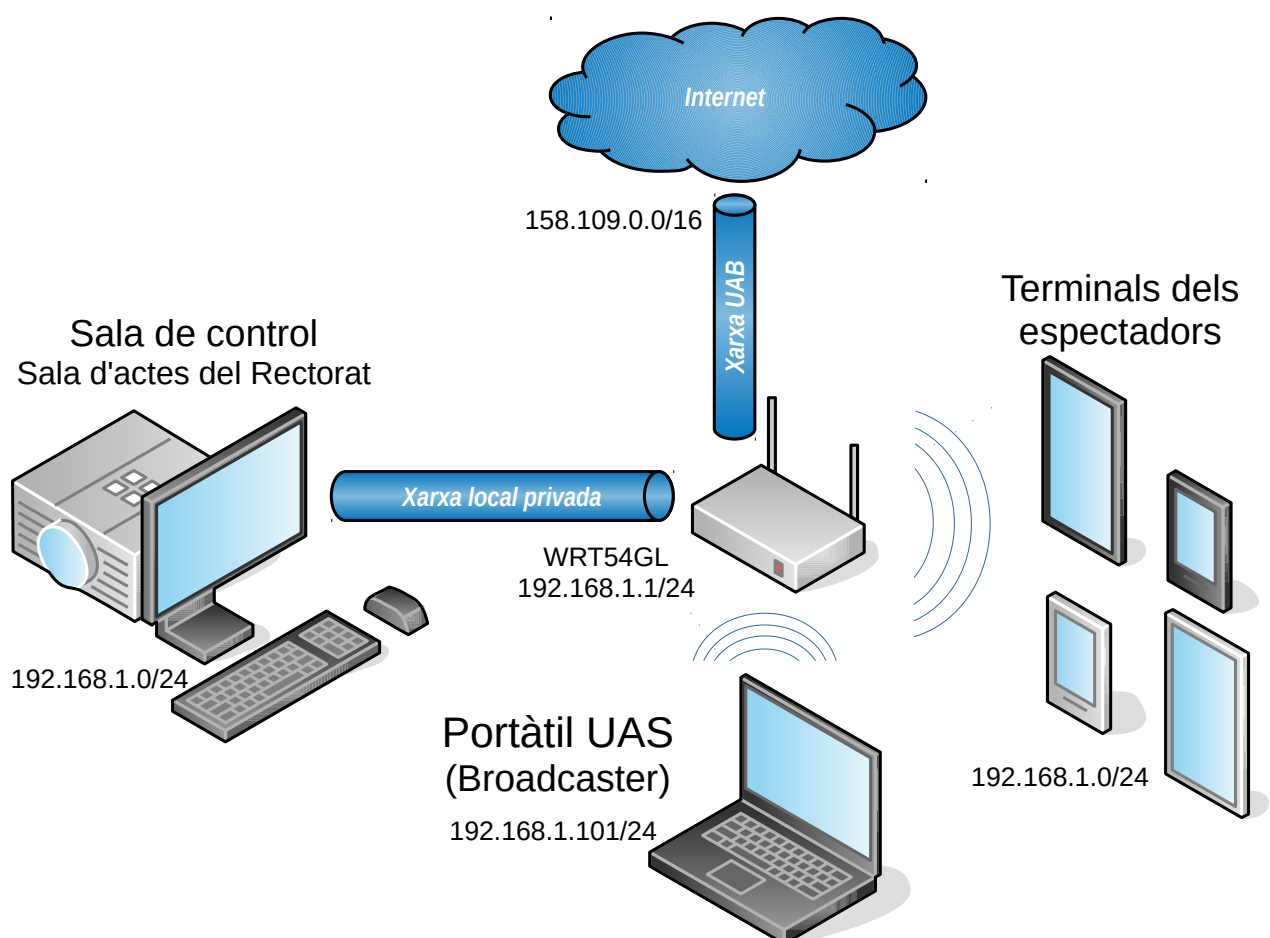


Figura 9: Diagrama físic del sistema

S'ha plantejat l'ús de *Multicast*, ja que en un futur, IPv6 no permetrà cap tipus de *Broadcast*, però per tal de simplificar la programació, i no haver de dependre d'encaminadors que hagin de donar suport a IGMP Snooping ni haver de donar d'alta i baixa els diversos clients, hem optat per a seguir fent servir les capacitats de *Broadcast* d'IPv4.

El codi font del programari, el seu contingut i les seves comunicacions estan codificades en Unicode, més concretament UTF-8, per tal de poder fer servir el software en qualsevol lloc del món o en qualsevol codificació. Per tant, atès que farem servir una xarxa sense fils, mòbil i temporal, i la seguretat no és un requeriment, enviarem els missatges en UDP i en clar.

La comunicació entre el client i el servidor es realitza en dues etapes simultànies:

- Per una banda, el servidor escolta peticions dels clients preguntant quins idiomes s'ofereixen i respon a cada client per separat.
- Per altra banda, el servidor es dedica a enviar els subtítols i les audiodescripcions independentment de quins clients hi hagi, a raó d'un idioma per port.

Servidor Broadcaster

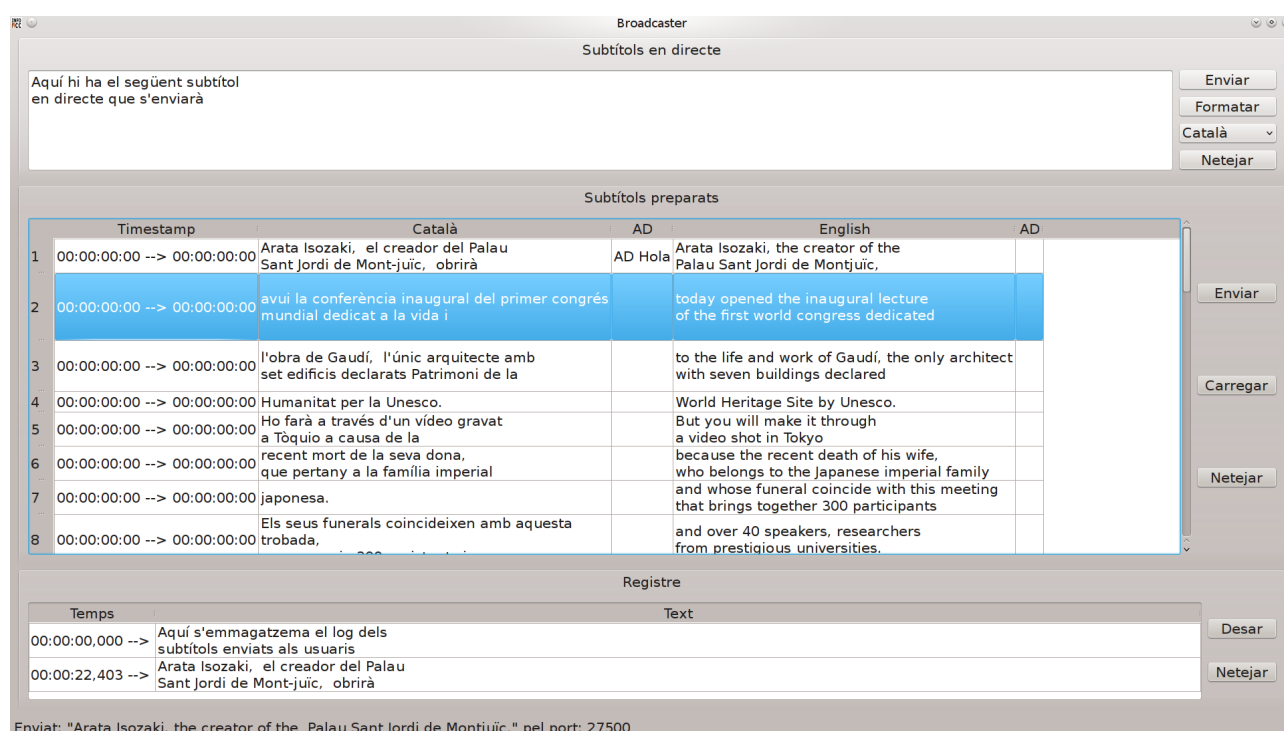


Figura 10: Servidor Broadcaster, programa principal del sistema

El servidor es presenta com una finestra amb tres apartats clarament diferenciats:

- Subtítols en directe
 - Conté un camp d'introducció de text que intenta mantenir el focus sempre.

- Un botó per a enviar els continguts (tecla F2)
- Un altre botó anomenat «Formatar» que divideix les frases llargues en múltiples línies (també F1), dividint-les quan sobrepassen els 30 caràcters i evitant dividir-les després de preposicions.
- Un selector d'idioma que permet escollir-ne entre cinc idiomes preestablerts.
- Un botó per a netejar el camp de text (també ESC).

Definir cinc idiomes ha estat un criteri escollit arbitràriament ja que inicialment es volia donar un accés compatible amb la ISO 3166-1 alpha-2¹⁰ però ja que l'aplicació pràctica del nostre projecte es limitava a molt pocs idiomes i de moment com a *proof of concept*, doncs hem donat espai per a cinc idiomes fixes: Català, castellà, anglès, francès i un emplaçament genèric anomenat «altres».

Aquesta primera finestra disposarà d'un *socket* UDP escoltant un port específic esperant rebre missatges codificats en XML sobre el reconeixedor automàtic de la parla en català (ASR) i automàticament els afegirà al final del text o on estigui el cursor dins del camp.

- Subtítols preparats
 - El segon conjunt d'opcions s'enfoca en esdeveniments on tenim informació prèvia, d'aquesta forma es disposa d'un botó per a carregar fitxers de subtítols en format SubRip Text (SRT) modificats lleugerament per a contenir les audiodescripcions que es volguin sintetitzar a veu per a les persones cegues, d'aquesta forma si una línia comença per «AD», l'ordinador entendre que no es un subtítol qualsevol i llegirà la línia a la persona cega.

La generació d'aquest fitxer es pot fer amb qualsevol programa d'edició de subtítols així que no ha calgut crear cap solució específica.

Una vegada seleccionat el fitxer, se'ns preguntarà de quin idioma es tracta i això omplirà la taula amb el contingut processat, si volem disposar de més idiomes només hem de carregar un altre fitxer i informar de quin idioma es tracta. Per tal d'evitar problemes, el fitxer haurà de contenir el mateix número de subtítols que els fitxers prèviament carregats i òbviament haurà de ser d'un idioma diferent als ja carregats.

10 ISO 3166 – Country codes – ISO. http://www.iso.org/iso/country_codes.htm

- Per a emetre els continguts podrem seleccionar qualsevol fila i prémer el botó d'enviar corresponent o fer servir F3 com a accés directe. Cada cop que enviem una línia, el cursor avança automàticament a la següent, simplificant molt la tasca.
- Finalment, si ens hem equivocat al carregar el fitxer o volem eliminar la informació preparada, disposem d'un botó de netejar taula.
- Registre
 - El darrer dels tres camps és el registre, aquí s'emmagatzema tot el que s'emet en directe i quan s'emet, també es desen els subtítols emesos del primer idioma que s'ha carregat des d'un fitxer.
 - La informació es pot desar en format SRT amb el botó corresponent.
 - Es disposa d'un botó per a buidar el registre en qualsevol moment.

Per a provar el funcionament de les emissions dels subtítols hem fet servir programes com el tcpdump i el wireshark, també lliures.

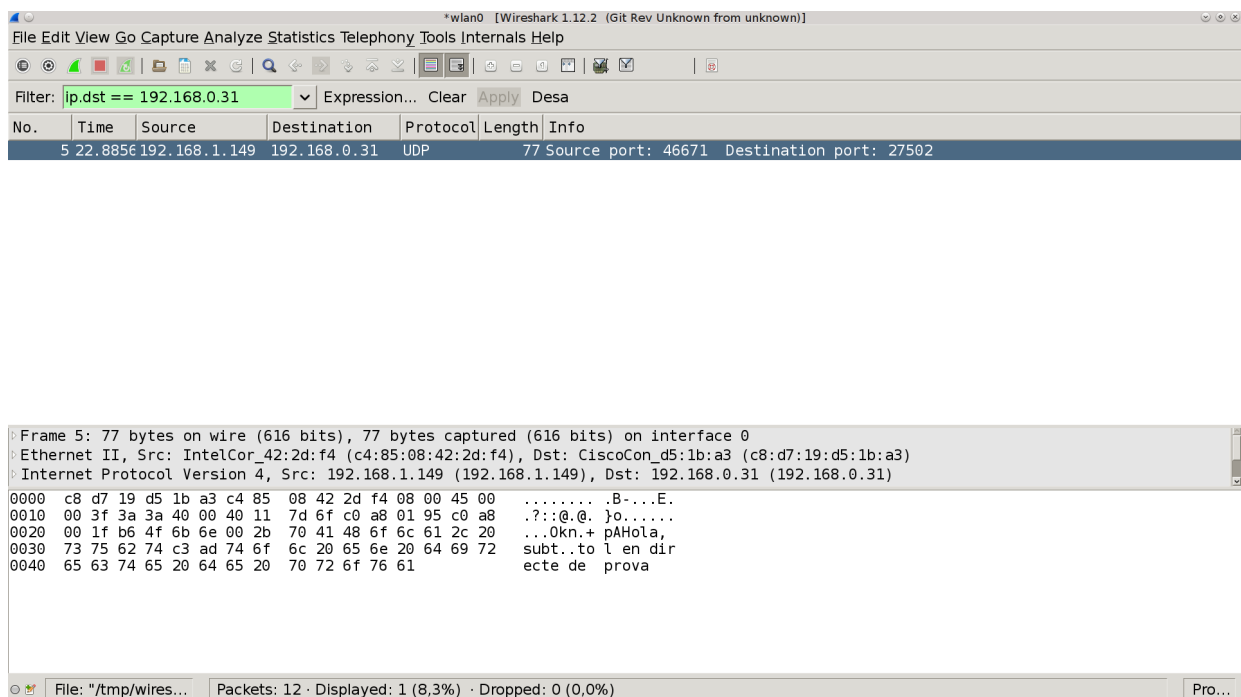


Figura 11: Wireshark, eina bàsica en la comprovació del funcionament en xarxa

Client Display

Existeixen dues maneres de programar una interfície gràfica en Qt, fent servir el mateix editor interactiu del Qt Creator i generar un fitxer XML (extensió *.ui) que serà posteriorment compilat i afegit al codi font, o programar directament amb codi font. Aquest programa està basat en el darrer mètode mentre que el primer s'ha posat en pràctica en el servidor.

Abans de programar l'aplicació mòbil, s'ha dissenyat també una aplicació Qt que faci part de les funcionalitats que se l'hi demanen al mòbil, és a dir, escollir i rebre subtítols en qualsevol dels idiomes.

Es tracta d'una aplicació amb dues parts clarament diferenciades, per una banda tenim una finestra que sempre està per sobre de tot on es mostra el contingut rebut per la xarxa, i d'altra banda una finestra que presenta un panell de control per a l'operador.



Figura 12: Display, el programa per a projectar els subtítols a una sala o capturar-los

Aquesta aplicació ha d'estar dins de la mateixa subxarxa que l'emissor (dins de la mateixa àrea de *Broadcast*) i que podem fer doble clic sobre la primera finestra per a cobrir qualsevol pantalla.

El compacte panell de control que ens permet seleccionar l'idioma que volem rebre, la tipografia del subtítol, la mida de la font, el color de la font i del fons. Aquest panell de control també disposa d'un camp de text que permet que l'operari que s'encarrega del canó pugui dirigir-se a la sala sobreposant-se a les emissions del servidor.

Aquesta aplicació és ideal per a poder incorporar els subtítols a una sala com el rectorat de la UAB on es disposa d'un equipament específic en el qual ens hi hem hagut d'adaptar.



Figura 13: Funcionament del programa dins de la sala

Client d'Android

S'ha fet una aplicació mòbil fent servir les plantilles de Google modernes, amb disseny Holo Light¹¹ per tal de donar un estil agradable i consistent amb la plataforma Android. Per a mostrar els subtítols, una pantalla fosca és un requeriment i per tant fem servir Holo Dark en pantalla completa.

S'han fet diversos *mockups* de dissenys per a fer més intuïtiu el funcionament del programa, el resultat és una aplicació minimalista amb només dos botons: subtítols i audiodescripció, que al clicar-los pregunta a una IP concreta quins idiomes ofereix.

¹¹ Themes | Android Developers, <http://developer.android.com/design/style/themes.html>

Es va proposar muntar una xarxa amb DNS per tal de fer el sistema molt més robust i adequat però es va acabar optant per a seguir amb el sistema fràgil d'adreces IP específiques. Un sistema molt senzill i fàcil que és viable només per que la xarxa és portàtil i la controlem nosaltres mateixos.

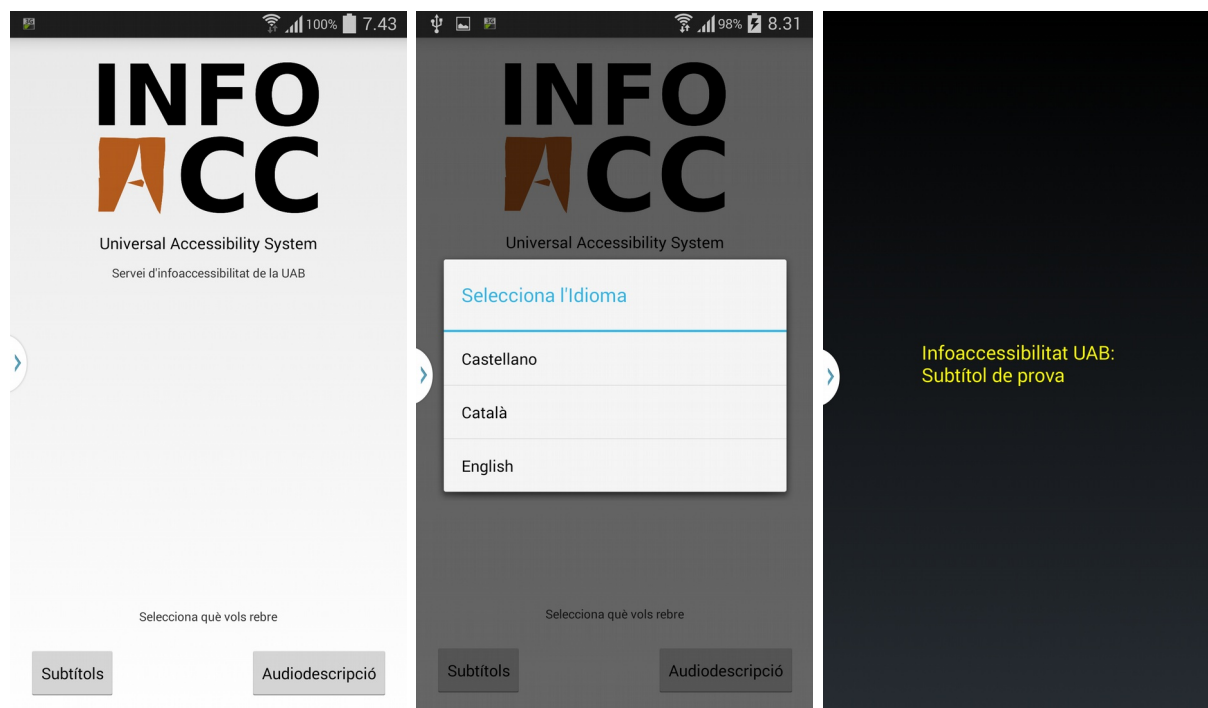


Figura 14: Programa d'Android Infoaccessibilitat UAB

Si el client no rep resposta en aproximadament un o dos segons, informa a l'usuari de que potser no està connectat a la xarxa wifi adequada, d'altra banda, en cas contrari, es mostra una llista d'idiomes que l'hi permetrà activar el sistema sense cap més intervenció.

Dins de la pantalla fosca de subtítols podrà decidir activar o desactivar la parla automàtica en funció dels seus gustos o necessitats.

Per a que l'Android sintetitzi veu en català fem servir una aplicació de pagament disponible al Google Play anomenada Acapella TTS amb la veu en català de la «laia».

Protocol

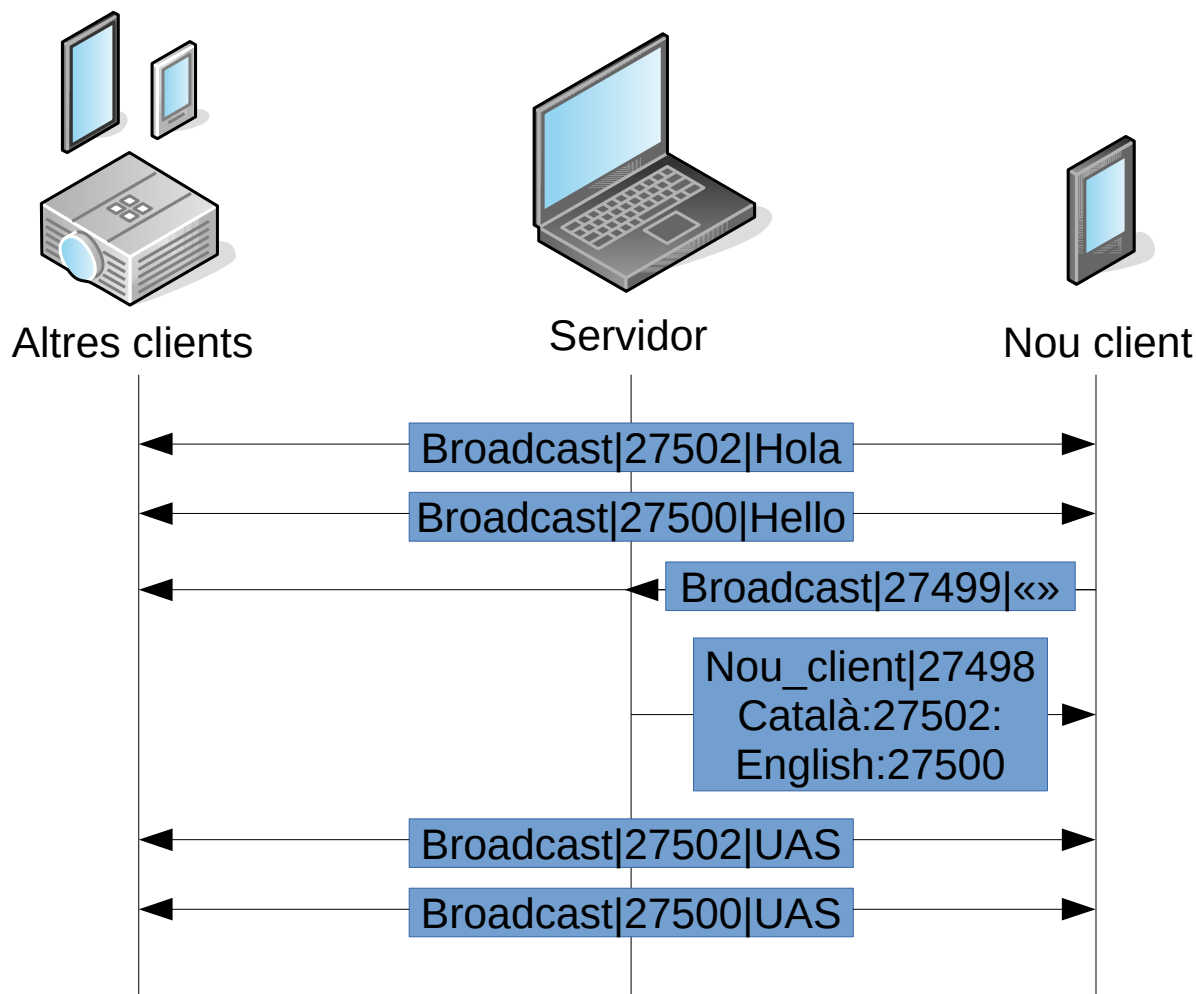


Figura 15: Protocol de reconeixement d'idiomes oferts

En aquest esquema de comunicació en xarxa es pot observar com el servidor emet els subtítols tant bon punt se l'hi ordena per part de l'operari i mentre es fa l'emissió, qualsevol client pot preguntar al mateix servidor quins idiomes disposa enviant un datagrama UDP a l'adreça de Broadcast, llavors el servidor l'hi respondrà amb un datagrama dirigit (*unicast*) al client que ho ha demanat informant dels idiomes disponibles així com el port on es poden escoltar.

Es fan servir multitud de ports, del 27500 al 27505 per als idiomes, el 27499 per a rebre les peticions d'idioma, el 27498 per a respondreles i el 27600 per a rebre la informació de l'ASR.

En les proves que s'han fet, 192.168.1.255 és l'adreça de Broadcast, 192.168.1.101 l'adreça del servidor proporcionada per reserva DHCP i la resta de dispositius obtenen adreces de la 102 a la 254. L'encaminador té connexió WAN amb DHCP a la UAB on obté una IP del rang 158.109.0.0/16 i d'aquesta forma podem proporcionar accés a Internet a qui ho necessiti si fos necessari.

5 Conclusions i ampliacions

S'han complert tots els objectius plantejats al començament d'aquesta memòria en quant al disseny i funcionalitats del software. La part que més temps ha portat ha estat aprendre a programar en Qt i Android, però malgrat tot, s'ha pogut complir les fites i acabar la feina en el temps previst. El sistema s'ha mostrat completament funcional davant del gerent de la UAB el 30 d'octubre de 2014 per a continuació convertir-lo a PyQt i esperar als resultats del seu funcionament a més llarg termini

Aquest projecte proporciona una eina fàcil i senzilla que permet acostar els esdeveniments en directe a les persones amb diversitat funcional, és per això que no s'ha buscat característiques avançades com ara modificadors de textos (negreta, cursiva), colors diferents (per a poder separar diferents ponents), etc. Aquestes modificacions no suposarien un gran esforç per a la implementació i es podrien implementar afegint modificadors als missatges que s'envien actualment o enviant metadades en una cadena XML per exemple. Malgrat això, el benefici que podria aportar al no és molt significatiu i per això no s'ha implementat.

Aquest projecte ha servit per a posar en pràctica els coneixements adquirits en diverses assignatures de l'enginyeria, fonamentalment xarxes de computadors, programació i estructures de dades.



Figura 16: Demostració de funcionament al gerent de la UAB al rectorat

Una millora substancial per al sistema seria definir un servidor DNS lleuger de forma que els dispositius mòbils contactessin amb el servidor a través del nom del host i no d'una IP fixa programada a l'aplicació mòbil. A més a més es podria fer servir Multicast per a cada conjunt d'idiomes que es requereixi emetre i així evitar emetre idiomes que ningú està interessat en escoltar.

Aquesta millora permetria generar estadístiques d'ús al servidor que permetin saber en tot moment qui rep què i de quina forma.

Una altra millora important seria definir un protocol que només faci servir un o dos ports de xarxa ja que l'IETF demana que les aplicacions facin servir el mínim nombre de ports possibles per a facilitar la usabilitat i convivència del gran nombre de programes existent avui dia, però nosaltres hem fet servir un port per a cada idioma per motius de compatibilitat amb aplicacions preexistents.

Tot i que el funcionament del sistema s'ha pogut provar en diverses ocasions a la sala d'actes del rectorat de la UAB, encara queda per veure quina utilitat suposa en un cas real.

Aquest projecte té una complexitat tècnica molt senzilla però una utilitat social elevada, és per això que aquesta memòria exposa més contingut funcional que tècnic.

Aquest projecte no hagués estat possible sense la formació rebuda al llarg de la carrera i en especialment a totes les assignatures relacionades directament amb la programació, a l'enginyeria del software, sistemes operatius i xarxes de computadors. Les altres assignatures m'han donat coneixements que en menor mesura també han influït.

Finalment vull agrair a tots els companys que han ajudat a fer possible aquest projecte: al dos tècnics en mitjans audiovisuals al rectorat (Jordis) , al Manel Ebri com a cap del servei, a l'ajuda del doctorand Héctor Delgado per als consells donats al llarg del projecte, i sobretot al mateix director, Javier Serrano, per la confiança i paciència que ha demostrat al llarg de tot el temps.

Bibliografia

Pino Lopes, Oscar (2011). *Accessibility Project for Theaters using Mobile Devices*. (Treball de fi de màster). Departament de Telecomunicació i Enginyeria de Sistemes. Universitat Autònoma de Barcelona. Bellaterra

Oncins Noguera, Estel·la (2013). *Accessibility for the scenic arts*. (Tesi doctoral). Departament de traducció i interpretació. Universitat Autònoma de Barcelona. Bellaterra

Referència de Qt 4.8: <http://qt-project.org/doc/qt-4.8/>

Referència d'Android: <http://developer.android.com/reference/packages.html>

Curs de programació en Android a la dEIC: <http://deic.uab.es/docencia/viewprog.php?idioma=0&codias=99002-0&style=>

Solucions Qt a Stack Overflow: <http://stackoverflow.com/questions/tagged/qt>

Solucions Android a Stack Overflow: <http://stackoverflow.com/questions/tagged/android>

Diari oficial de la Generalitat de Catalunya: <http://dogc.gencat.cat/ca>

Vídeos de la UAB: <http://www.uab.cat/videos/>

Resum en català

Disseny i implementació d'un sistema de retransmissió de continguts accessibles, subtítols i audiodescripcions, a través del dispositius mòbils dels mateixos usuaris. La informació s'envia en datagrames Broadcast UDP i els continguts es poden generar en directe a través d'un reconeixedor de parla en múltiples idiomes (en català es fa servir un software programat a la UAB). El sistema és de molt baix cost i molt fàcilment ampliable. Les audiodescripcions i els subtítols poden ser llegits pel sistema de *text-to-speech* del mòbil per a les persones cegues o que vulguin escoltar un altre idioma.

Resum en castellà

Diseño y implementación de un sistema de retransmisión de contenidos accesibles, subtítulos y audiodescripciones, a través de los dispositivos móviles de los mismos usuarios. La información se envía en datagramas Broadcast UDP y los contenidos se pueden generar en directo a través de un reconocedor de habla en múltiples idiomas (en catalán se usa un software programado en la UAB). El sistema es de muy bajo coste y muy fácilmente ampliable. Las audiodescripciones y los subtítulos pueden ser leídos por el sistema de *text-to-speech* del móvil para las personas ciegas o que quieran escuchar otro idioma.

Resum en anglès

Design and implementation of an accessible content broadcasting system, mainly subtitles and audio descriptions, using the users own mobile devices. The information is sent using Broadcast UDP datagrams and their contents can be generated live using a multilingual automated speech recognizer (Catalan language uses UAB programmed software). The system is very low cost and easy to expand. The audio descriptions and subtitles can be read by the smartphones own text-to-speech system at blind people and people who would like to listen a different language.